

CLAIMS

What is claimed is:

- 1 1. An apparatus comprising:
 - 2 a priority register to store thread information for P threads, the thread
 - 3 information including P priority codes corresponding to the P threads, at least one
 - 4 of the P threads requesting use of at least one resource unit; and
 - 5 a priority selector coupled to the priority register to generate assignment
 - 6 signal to assign the at least one resource unit to the at least one of the P threads
 - 7 according to the P priority codes.
- 1 2. The apparatus of claim 1 wherein the at least one resource unit is
- 2 one of an instruction fetch unit, an instruction buffer, a memory locking unit, a
- 3 load unit, a store unit, an input/output unit, a peripheral unit interface, and a
- 4 functional unit.
- 1 3. The apparatus of claim 2 wherein the functional unit is one of an
- 2 arithmetic unit, a logic unit, and an arithmetic and logic unit.
- 1 4. The apparatus of claim 1 further comprising:
 - 2 an instruction multiplexer coupled to the priority selector to pass
 - 3 instructions stored in a plurality of instruction registers to execution units
 - 4 according to the assignment signal.

- 1 5. The apparatus of claim 1 further comprising:
2 a priority assignor coupled to the priority register to set the thread
3 information including at least one of the P priority codes corresponding to the at
4 least one of the P threads in response to a start instruction from an instruction
5 decoder and dispatcher.
- 1 6. The apparatus of claim 5 wherein the priority assignor sets an
2 active flag in the priority register corresponding to the at least one of the P threads
3 in response to the start instruction.
- 1 7. The apparatus of claim 6 wherein resets the active flag in the
2 priority register corresponding to the at least one of the P threads in response to a
3 quit instruction from the instruction decoder and dispatcher.
- 1 8. The apparatus of claim 1 wherein the priority selector assigns the at
2 least one resource unit to the at least one of the P threads if the at least one of the
3 P threads is not served and the at least one resource unit is free.
- 1 9. The apparatus of claim 8 wherein the at least one of the P threads
2 has highest priority code among a set of ready threads of the P threads.
- 1 10. The apparatus of claim 8 wherein the priority selector iteratively
2 assigns resource units to threads in the set of ready threads of the P threads
3 according to the corresponding priority codes and resource availability until the set
4 becomes empty.

1 11. A method comprising:
2 storing thread information for P threads, the thread information including
3 P priority codes corresponding to the P threads, at least one of the P threads
4 requesting use of at least one resource unit; and
5 generating assignment signal to assign the at least one resource unit to the
6 at least one of the P threads according to the P priority codes.

1 12. The method of claim 11 wherein the at least one resource unit is
2 one of an instruction fetch unit, an instruction buffer, a memory locking unit, a
3 load unit, a store unit, an input/output unit, a peripheral unit interface, and a
4 functional unit.

1 13. The method of claim 12 wherein the functional unit is one of an
2 arithmetic unit, a logic unit, and an arithmetic and logic unit.

1 14. The method of claim 11 further comprising:
2 passing instructions stored in a plurality of instruction registers to
3 execution units according to the assignment signal.

1 15. The method of claim 11 further comprising:
2 setting the thread information including at least one of the P priority codes
3 corresponding to the at least one of the P threads in response to a start instruction
4 from an instruction decoder and dispatcher.

1 16. The method of claim 15 wherein setting the thread information
2 comprises setting an active flag in the priority register corresponding to the at least
3 one of the P threads in response to the start instruction.

1 17. The method of claim 16 wherein setting the thread information
2 comprises resetting the active flag in the priority register corresponding to the at
3 least one of the P threads in response to a quit instruction from the instruction
4 decoder and dispatcher.

1 18. The method of claim 11 wherein generating the assignment signal
2 comprises generating the assignment signal to assign the at least one resource unit
3 to the at least one of the P threads if the at least one of the P threads is not served
4 and the at least one resource unit is free.

1 19. The method of claim 18 wherein the at least one of the P threads
2 has highest priority code among a set of ready threads of the P threads.

1 20. The method of claim 1 wherein generating the assignment signal
2 comprises iteratively assigning resource units to threads in the set of ready threads
3 of the P threads according to the corresponding priority codes and resource
4 availability until the set becomes empty.

1 21. A processor comprising:
2 at least one a resource unit to provide resource for use by P threads; and

3 a resource prioritizer coupled to the resource unit to prioritize resource
4 utilization, the resource prioritizer comprising:

5 a priority register to store thread information for the P threads, the
6 thread information including P priority codes corresponding to the
7 P threads, at least one of the P threads requesting use of the at least
8 one resource unit, and

9 a priority selector coupled to the priority register to generate
10 assignment signal to assign the at least one resource unit to the at
11 least one of the P threads according to the P priority codes.

1 22. The processor of claim 21 wherein the at least one resource unit is
2 one of an instruction fetch unit, an instruction buffer, a memory locking unit, a
3 load unit, a store unit, an input/output unit, a peripheral unit interface, and a
4 functional unit.

1 23. The processor of claim 22 wherein the functional unit is one of an
2 arithmetic unit, a logic unit, and an arithmetic and logic unit.

1 24. The processor of claim 21 the resource prioritizer further
2 comprising:

3 an instruction multiplexer coupled to the priority selector to pass
4 instructions stored in a plurality of instruction registers to execution units
5 according to the assignment signal.

1 25. The processor of claim 21 wherein the resource prioritizer further
2 comprising:

3 a priority assignor coupled to the priority register to set the thread
4 information including at least one of the P priority codes corresponding to the at
5 least one of the P threads in response to a start instruction from an instruction
6 decoder and dispatcher.

1 26. The processor of claim 25 wherein the priority assignor sets an
2 active flag in the priority register corresponding to the at least one of the P threads
3 in response to the start instruction.

1 27. The processor of claim 26 wherein resets the active flag in the
2 priority register corresponding to the at least one of the P threads in response to a
3 quit instruction from the instruction decoder and dispatcher.

1 28. The processor of claim 21 wherein the priority selector assigns the
2 at least one resource unit to the at least one of the P threads if the at least one of
3 the P threads is not served and the at least one resource unit is free.

1 29. The processor of claim 28 wherein the at least one of the P threads
2 has highest priority code among a set of ready threads of the P threads.

1 30. The processor of claim 28 wherein the priority selector iteratively
2 assigns resource units to threads in the set of ready threads of the P threads
3 according to the corresponding priority codes and resource availability until the set
4 becomes empty.